EXPERIMENT - 11

Object Oriented Programming Lab

Aim

Write a program to demonstrate the use of friend function with Inline assignment.

Syeda Reeha Quasar

14114802719
4C7

EXPERIMENT – 11

Aim:

Write a program to demonstrate the use of friend function with Inline assignment.

Source Code:

```cpp
#include <iostream>
using namespace std;

class threeNumbers{
    private:
        int x, y, z;

    public:
        void input(){
            cout << "Enter three numbers: ";
            cin >> x >> y >> z;
        }

        friend inline void findLargest(threeNumbers t);
};

inline void findLargest(threeNumbers t){
    if (t.x > t.y && t.x > t.z) {
        cout << "Largest is:" << t.x;
    }
    else if (t.y > t.z) {
        cout << "Largest is:" << t.y;
    }
```

```cpp
    else {
        cout << "Largest is:" << t.z;
    }
}


int main(){
    threeNumbers t;
    t.input();
    findLargest(t);
    return 0;
}
```

Output:

```
PS D:\sem 4\cpp\oops> cd "d:\sem 4\cpp\oops\" ; if ($?) { g++ inlineFriend.cpp -o inlineFriend } ; if ($?) { .\inlineFriend }
Enter three numbers: 23 45 32
Largest is:45
PS D:\sem 4\cpp\oops> cd "d:\sem 4\cpp\oops\" ; if ($?) { g++ inlineFriend.cpp -o inlineFriend } ; if ($?) { .\inlineFriend }
Enter three numbers: 23 45 34
Largest is:45
PS D:\sem 4\cpp\oops> cd "d:\sem 4\cpp\oops\" ; if ($?) { g++ inlineFriend.cpp -o inlineFriend } ; if ($?) { .\inlineFriend }
Enter three numbers:
12
4
23
Largest is:23
PS D:\sem 4\cpp\oops>
```

```
PS D:\sem 4\cpp\oops> cd "d:\sem 4\cpp\oops\" ; if ($?) { g++ inlineFriend.cpp -o inlineFriend } ; if ($?) { .\inlineFriend }
Enter three numbers: 2 4 5
Largest is:5
```

Viva Questions

Q1). What are friend functions?
Ans.

Friend Functions:

If a function is defined as a friend function then, the private and protected data of a class can be accessed using the function.

The complier knows a given function is a friend function by the use of the keyword friend.

For accessing the data, the declaration of a friend function should be made inside the body of the class (can be anywhere inside class either in private or public section) starting with keyword friend.

Q2). What are characteristics of friend function?
Ans.

Characteristics of friend function are as follows:

- friend function is not in the scope of the class in which it has been declared as friend.

- It cannot be called using the object of the class as it is not in the scope of the class.

- It can be called similar to a normal function without the help of any object.

- Unlike member functions, a friend function cannot access the member variables directly and has to use an object name and dot membership operator with each member variable.

- It can be declared either in the public or private scope area of a class.

- Usually, it has objects as arguments.

Q4. What are the benefits of friend function?
Ans.

- A friend function is used to access the non-public members of a class.

- It allows to generate more efficient code.

- It provides additional functionality which is not normally used by the class.

- It allows to share private class information by a non member function.

- It is used when two or more classes may contain members that are interrelated relative to others parts of the program.

Q5. What are disadvantages of friend function?
Ans.

- A derived class does not inherit friend function.

- Friend functions can not have a storage class specifier i.e they can not be declared as static or extern.